Introduction

In this report, we are covering the fundamentals of the data analysis process that was used for malware classification. The objective is divided into two primary goals: The first is to understand measures of statistical tendencies contained within the raw data. In doing so, we will ensure a clear understanding of the hidden mechanisms contained and the statistical underpinnings of the information.

The second objective is to propose a strategy for breaking down the data for use within the scope of the machine learning research proposed in the previously-discussed paper. The strategy will be justified by the learned tendencies, as well as by trial-and-error as determined by observation.

Dataset Shape

We first observe the structure of the dataset as demonstrated by the following machine output:

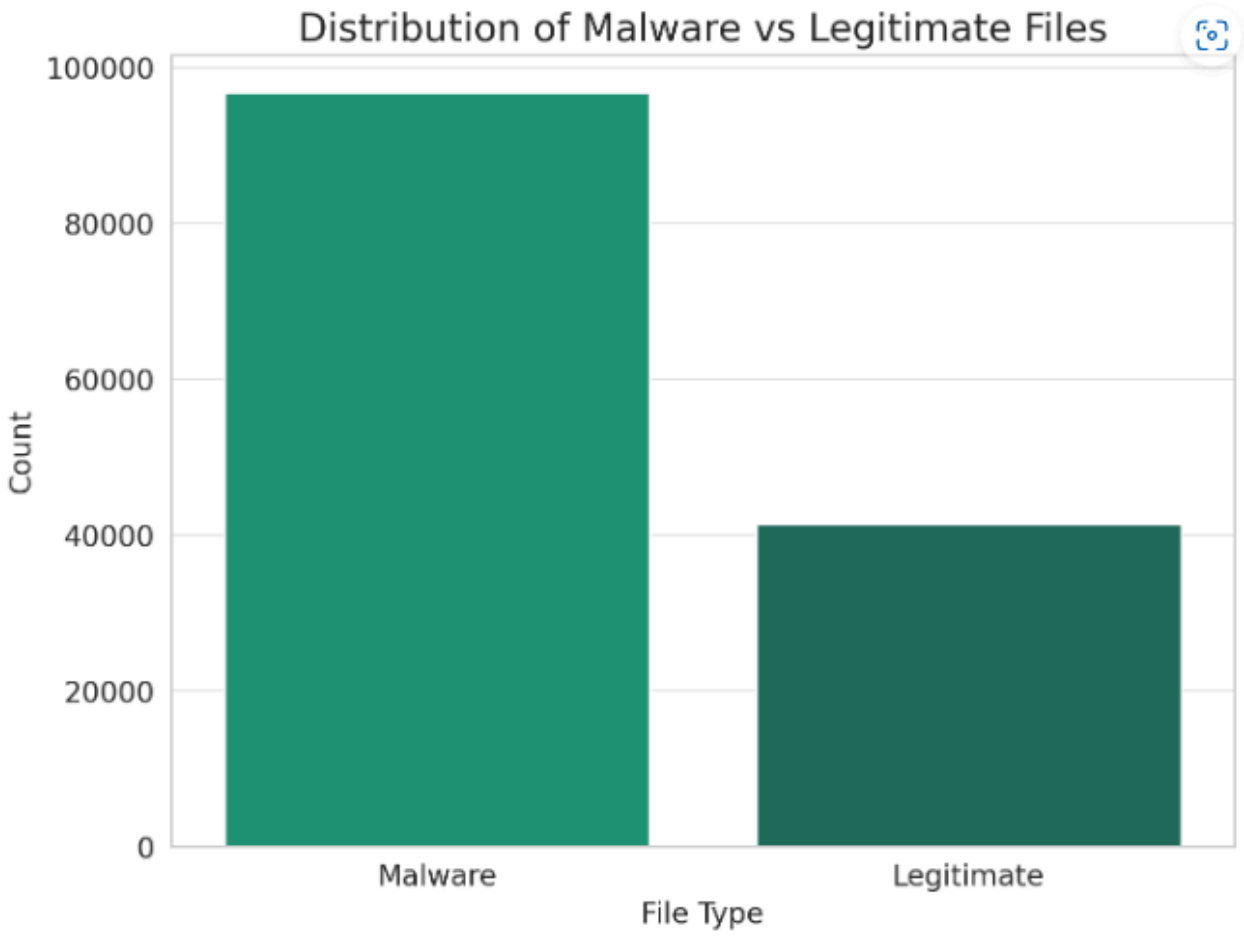
```
           Name                                md5  Machine  \
0   memtest.exe  631ea355665f28d4707448e442fbf5b8      332
1       ose.exe  9d10f99a6712e28f8acd5641e3a7ea6b      332
2     setup.exe  4d92f518527353c0db88a70fddcfd390      332
3      DW20.EXE  a41e524f8d45f0074fd07805ff0c9b12      332
4  dwtrig20.exe  c87e561258f2f8650cef999bf643a731      332

   SizeOfOptionalHeader  Characteristics  MajorLinkerVersion  \
0                   224              258                   9
1                   224             3330                   9
2                   224             3330                   9
3                   224              258                   9
4                   224              258                   9

   MinorLinkerVersion  SizeOfCode  SizeOfInitializedData  \
0                   0      361984                 115712
1                   0      130560                  19968
2                   0      517120                 621568
3                   0      585728                 369152
4                   0      294912                 247296

   SizeOfUninitializedData  ...  ResourcesNb  ResourcesMeanEntropy  \
0                        0  ...            4              3.262823
1                        0  ...            2              4.250461
2                        0  ...           11              4.426324
3                        0  ...           10              4.364291
4                        0  ...            2              4.306100

   ResourcesMinEntropy  ResourcesMaxEntropy  ResourcesMeanSize  \
0             2.568844             3.537939        8797.000000
```

```
1              3.420744             5.080177         837.000000
2              2.846449             5.271813       31102.272727
3              2.669314             6.400720        1457.000000
4              3.421598             5.190603        1074.500000

   ResourcesMinSize  ResourcesMaxSize  LoadConfigurationSize  \
0               216             18032                      0
1               518              1156                     72
2               104            270376                     72
3                90              4264                     72
4               849              1300                     72

   VersionInformationSize  legitimate
0                      16           1
1                      18           1
2                      18           1
3                      18           1
4                      18           1

[5 rows x 57 columns]
```

In total, we observe that there are 57 columns, of which, one, ‘legitimate’ is used for the ground truth of the experiment. Ground truth is extracted on the basis of VirusShare data.

The dataset consists of 152,227 samples of program metadata, of which 138,047 are recoverable. The total loss consists of 14,180 rows, which are discarded due to empty, missing, or incomplete data. A significant portion of the eliminated binaries are of malware samples. This is because they are extracted primary from datasets that might contain outdated or otherwise obsolete references.

Of the 138,047 remaining program metadata samples, 96,724 represent malware metadata and 41,323 represent legitimate program metadata. The distribution is approximately 2.3:1 as a ratio of malware program to legitimate program metadata. The percentages are demonstrated as:



It is clear that the primary composition of this dataset is Malware program metadata. Here we observe a concern regarding the differences between the true ratio of Malware:Legitimate program in the real world, versus what is observed here. This will be discussed in more detail in a proceeding section. The initial suspicion is that this will create a false-positive tendency where the models will lean towards incorrectly labeling legitimate programs as malware.

Data Shape

The data is described by the following code snippet:

```
Integer (int64): 45 columns
Floating-point (float64): 10 columns
String (object): 2 columns (Name and md5)
```

Of the 57 columns, 45 are represented strictly by integer values and 10 are represented by floating-point values. Here we observe another critical point: While the data is often suitably contained within integer-type variables, we cannot guarantee that there was no data loss, or that a floating-point value may have been more helpful for quantifying statistical tendencies. This will also be discussed further in a proceeding section.

Statistical Measures of Data

```
Machine  SizeOfOptionalHeader  Characteristics  \
count  138047.000000         138047.000000    138047.000000
mean     4259.069274            225.845632      4444.145994
std     10880.347245              5.121399      8186.782524
min       332.000000            224.000000         2.000000
25%       332.000000            224.000000       258.000000
50%       332.000000            224.000000       258.000000
75%       332.000000            224.000000      8226.000000
max     34404.000000            352.000000     49551.000000

       MajorLinkerVersion  MinorLinkerVersion    SizeOfCode  \
count       138047.000000       138047.000000  1.380470e+05
mean             8.619774            3.819286  2.425956e+05
std              4.088757           11.862675  5.754485e+06
min              0.000000            0.000000  0.000000e+00
25%              8.000000            0.000000  3.020800e+04
50%              9.000000            0.000000  1.136640e+05
75%             10.000000            0.000000  1.203200e+05
max            255.000000          255.000000  1.818587e+09

        SizeOfInitializedData  SizeOfUninitializedData  AddressOfEntryPoint
\
count           1.380470e+05             1.380470e+05         1.380470e+05
mean            4.504867e+05             1.009525e+05         1.719561e+05
std             2.101599e+07             1.635288e+07         3.430553e+06
min             0.000000e+00             0.000000e+00         0.000000e+00
25%             2.457600e+04             0.000000e+00         1.272100e+04
50%             2.631680e+05             0.000000e+00         5.288300e+04
75%             3.850240e+05             0.000000e+00         6.157800e+04
max             4.294966e+09             4.294941e+09         1.074484e+09

         BaseOfCode  ...    ResourcesNb  ResourcesMeanEntropy  \
count  1.380470e+05  ...  138047.000000         138047.000000
mean   5.779845e+04  ...      22.050700              4.000127
std    5.527658e+06  ...     136.494244              1.112981
min    0.000000e+00  ...       0.000000              0.000000
25%    4.096000e+03  ...       5.000000              3.458505
50%    4.096000e+03  ...       6.000000              3.729824
75%    4.096000e+03  ...      13.000000              4.233051
max    2.028711e+09  ...    7694.000000              7.999723
```

```
       ResourcesMinEntropy  ResourcesMaxEntropy  ResourcesMeanSize  \
count        138047.000000        138047.000000       1.380470e+05
mean              2.434541             5.521610       5.545093e+04
std               0.815577             1.597403       7.799163e+06
min               0.000000             0.000000       0.000000e+00
25%               2.178748             4.828706       9.560000e+02
50%               2.458492             5.317552       2.708154e+03
75%               2.696833             6.502239       6.558429e+03
max               7.999723             8.000000       2.415919e+09

       ResourcesMinSize  ResourcesMaxSize  LoadConfigurationSize  \
count      1.380470e+05      1.380470e+05           1.380470e+05
mean       1.818082e+04      2.465903e+05           4.656750e+05
std        6.502369e+06      2.124860e+07           2.608987e+07
min        0.000000e+00      0.000000e+00           0.000000e+00
25%        4.800000e+01      2.216000e+03           0.000000e+00
50%        4.800000e+01      9.640000e+03           7.200000e+01
75%        1.320000e+02      2.378000e+04           7.200000e+01
max        2.415919e+09      4.294903e+09           4.294967e+09

       VersionInformationSize     legitimate
count           138047.000000  138047.000000
mean                12.363115       0.299340
std                  6.798878       0.457971
min                  0.000000       0.000000
25%                 13.000000       0.000000
50%                 15.000000       0.000000
75%                 16.000000       1.000000
max                 26.000000       1.000000

[8 rows x 55 columns]
```

Observed are the general statistical tendencies measured for each of the columns of the dataset. The factors considered are the element counts, which is equal among all fields, statistical mean, standard deviation, minimum, and percentiles per column. An evaluation of the difference between mean, median, mode had little effect on the final analysis.

The most important statistic to obtain at this current stage is the correlation between every value and the ground truth. In the case of this dataset, the outcome is the following:

```
legitimate                1.000000
Machine                   0.548835
SizeOfOptionalHeader      0.547498
```

Subsystem	0.514352
MajorSubsystemVersion	0.380393
VersionInformationSize	0.379646
ResourcesMinEntropy	0.299112
Characteristics	0.221956
ExportNb	0.134408
ImportsNbOrdinal	0.128112
FileAlignment	0.125169
ImportsNb	0.116415
ResourcesNb	0.090405
MajorImageVersion	0.084410
MinorImageVersion	0.083220
SectionsMinRawsize	0.059346
SectionsMinVirtualsize	0.056466
ImportsNbDLL	0.038395
SizeOfCode	0.017476
MajorLinkerVersion	0.017320
SizeOfHeaders	0.010125
ImageBase	0.008245
MajorOperatingSystemVersion	0.002402
SectionsMeanVirtualsize	0.001734
SectionsMeanRawsize	0.001175
AddressOfEntryPoint	-0.000134
SectionMaxRawsize	-0.000790
BaseOfData	-0.001136
MinorSubsystemVersion	-0.001213
SectionMaxVirtualsize	-0.001332
MinorOperatingSystemVersion	-0.001702
ResourcesMinSize	-0.001774
SectionAlignment	-0.002429
SizeOfHeapCommit	-0.002506
SizeOfImage	-0.002603
LoaderFlags	-0.002649
SizeOfStackCommit	-0.003226
NumberOfRvaAndSizes	-0.003523
ResourcesMeanSize	-0.003824
SizeOfUninitializedData	-0.003997
SizeOfInitializedData	-0.004958
ResourcesMaxSize	-0.005529
BaseOfCode	-0.006232
LoadConfigurationSize	-0.011666
MinorLinkerVersion	-0.146652
SectionsMinEntropy	-0.152840

```
SizeOfHeapReserve              -0.156260
CheckSum                       -0.195329
ResourcesMeanEntropy           -0.202432
SectionsNb                     -0.207782
SectionsMeanEntropy            -0.343933
ResourcesMaxEntropy            -0.392855
SizeOfStackReserve             -0.521642
SectionsMaxEntropy             -0.624229
DllCharacteristics             -0.630177
Name: legitimate, dtype: float64
```

The legitimate flag, which serves as ground truth, has a correlation of 1 with itself. The following list is the 10 elements with highest correlation to ground truth excluding the legitimate tag:

```
Machine: 0.5488
SizeOfOptionalHeader: 0.5475
Subsystem: 0.5144
MajorSubsystemVersion: 0.3804
VersionInformationSize: 0.3796
ResourcesMinEntropy: 0.2991
Characteristics: 0.2220
ExportNb: 0.1344
ImportsNbOrdinal: 0.1281
FileAlignment: 0.1252
```

The following are the elements with the 10 lowest correlations to ground truth:

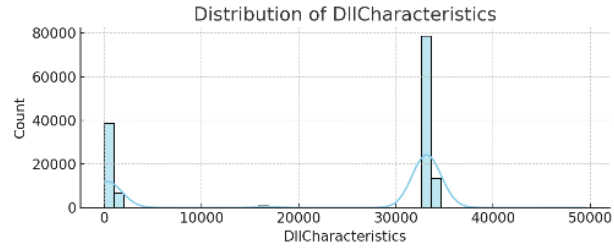
```
DllCharacteristics: -0.6302
SectionsMaxEntropy: -0.6242
SizeOfStackReserve: -0.5216
ResourcesMaxEntropy: -0.3929
SectionsMeanEntropy: -0.3440
SectionsNb: -0.2078
ResourcesMeanEntropy: -0.2024
CheckSum: -0.1953
SizeOfHeapReserve: -0.1563
SectionsMinEntropy: -0.1528
```

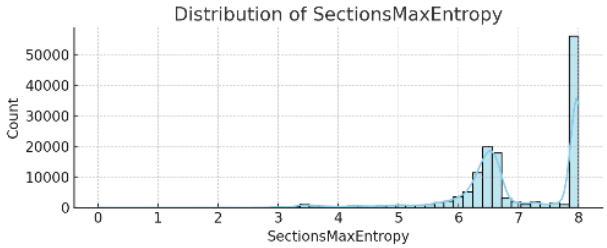
The following are the 20 elements with highest absolute value correlation to the ground truth:

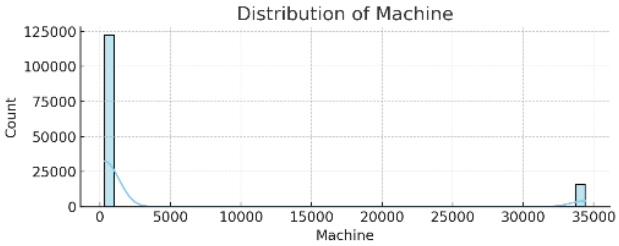
```
DllCharacteristics: 0.630177
SectionsMaxEntropy: 0.624229
Machine: 0.548835
SizeOfOptionalHeader: 0.547498
SizeOfStackReserve: 0.521642
Subsystem: 0.514352
```

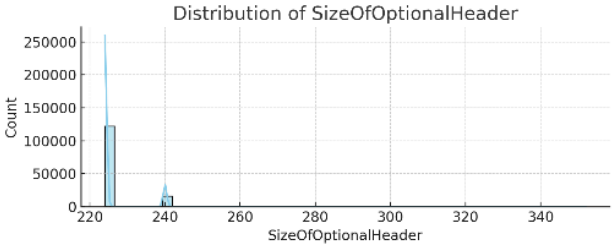
```
ResourcesMaxEntropy: 0.392855
MajorSubsystemVersion: 0.380393
VersionInformationSize: 0.379646
SectionsMeanEntropy: 0.343933
ResourcesMinEntropy: 0.299112
Characteristics: 0.221956
SectionsNb: 0.207782
ResourcesMeanEntropy: 0.202432
CheckSum: 0.195329
SizeOfHeapReserve: 0.156260
SectionsMinEntropy: 0.152840
MinorLinkerVersion: 0.146652
ExportNb: 0.134408
ImportsNbOrdinal: 0.128112
```

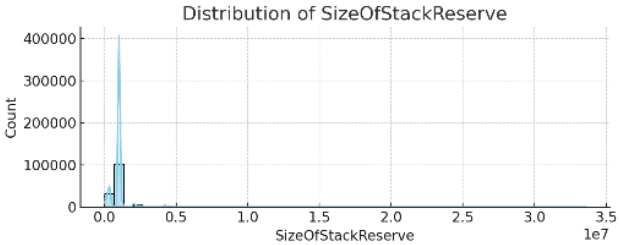
Below is a representation of each of these elements through histograms, whose range is relative to each column. The goal here is to identify tendencies of the data.

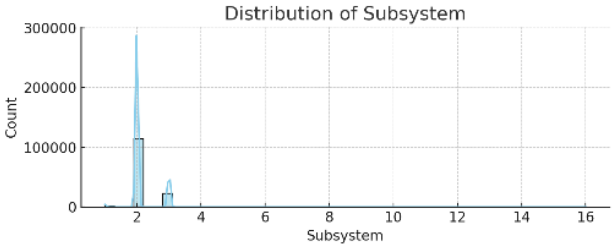

Distribution of DllCharacteristics
Count
DllCharacteristics

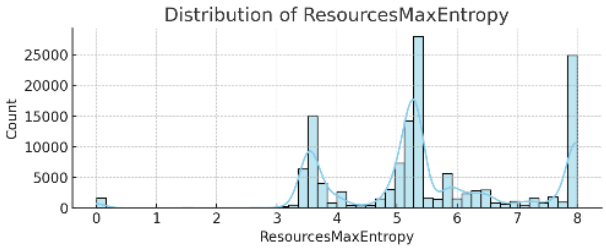

Distribution of SectionsMaxEntropy
Count
SectionsMaxEntropy

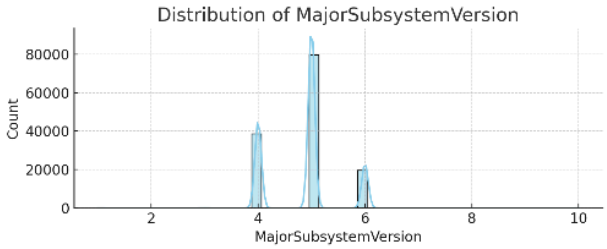

Distribution of Machine
Count
Machine

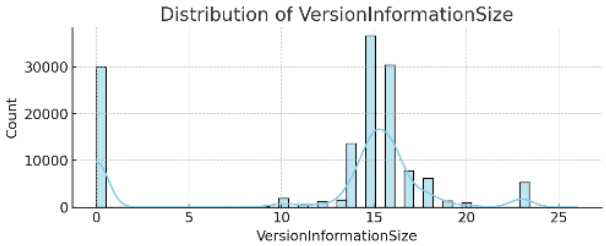

Distribution of SizeOfOptionalHeader
Count
SizeOfOptionalHeader

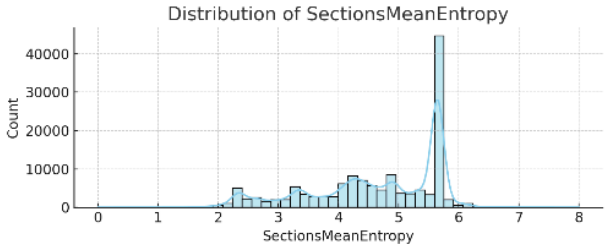

Distribution of SizeOfStackReserve
Count
SizeOfStackReserve
1e7

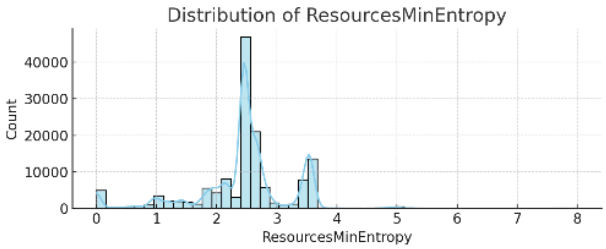

Distribution of Subsystem
Count
Subsystem

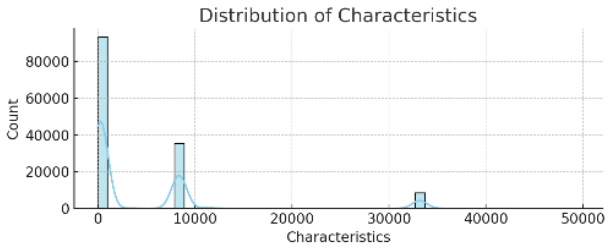

Distribution of ResourcesMaxEntropy
Count
ResourcesMaxEntropy

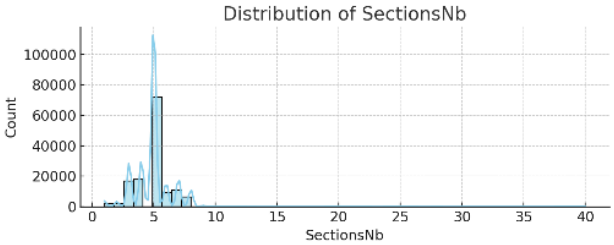

Distribution of MajorSubsystemVersion
Count
MajorSubsystemVersion

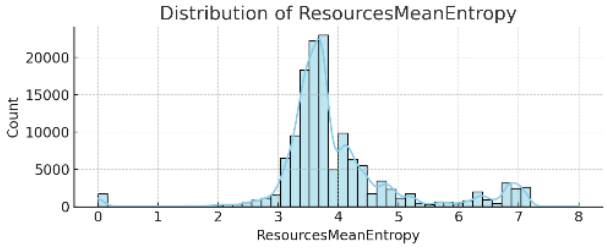

Distribution of VersionInformationSize
Count
VersionInformationSize


Distribution of SectionsMeanEntropy
Count
SectionsMeanEntropy

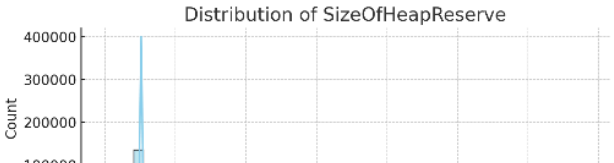

Distribution of ResourcesMinEntropy
Count
ResourcesMinEntropy


Distribution of Characteristics
Count
Characteristics


Distribution of SectionsNb
Count
SectionsNb


Distribution of ResourcesMeanEntropy
Count
ResourcesMeanEntropy


Distribution of CheckSum
Count


Distribution of SizeOfHeapReserve
Count

We observe unusual distributions in some of the more highly-correlated values, such as machine and DllCharacteristics. This point will be discussed in more detail in a proceeding section.

It's essential to understand roughly what each of the dataset values mean, so as to be able to reproduce the results with newly-generated data. Below is an explanation of the most important columns in the dataset, selected from the elements with high absolute value correlation to the ground truth:

Machine (Positive Correlation): This feature indicates the architecture of the target machine for which the binary was compiled (e.g., x86, x64, ARM). A hypothesis could be that certain architectures are more commonly associated with legitimate software than with malware, or vice versa.

SizeOfOptionalHeader (Positive Correlation): The size of the optional header can vary based on the format or version of the binary. Legitimate software might follow more standardized or recent formats, leading to a consistent size for the optional header.

Subsystem (Positive Correlation): This represents the environment in which the executable runs (e.g., GUI, console). Legitimate applications might predominantly use certain subsystems, while malware might target others.

SectionsMaxEntropy (Negative Correlation): Entropy is a measure of randomness or unpredictability. High entropy in a binary section can be indicative of obfuscation or encryption, techniques often used by malware to hide their code from static analysis.

DllCharacteristics (Negative Correlation): This represents certain flags or attributes set in the binary related to DLLs. Malware might use certain tricks or techniques that manifest in these characteristics to achieve persistence or evasion.

SizeOfStackReserve (Negative Correlation): This specifies the total size of memory to reserve for the stack. Malware might manipulate this value to either evade detection or to exploit certain vulnerabilities.

MajorSubsystemVersion (Positive Correlation): Indicates the major version number of the required subsystem. Legitimate software might be updated more frequently to use the latest subsystems, while malware might target older, more vulnerable subsystems.

ResourcesMaxEntropy (Negative Correlation): Similar to SectionsMaxEntropy, but specifically for resources. High entropy in resources might indicate that malware is embedding encrypted or obfuscated data.

The following is a brief overview of all metadata:

SizeOfOptionalHeader: Length of the optional header section, indicating the binary's format or version.
Characteristics: Flags that define characteristics of the executable, like type and execution state.
MajorLinkerVersion: Major version number of the linker used to create the executable.
MinorLinkerVersion: Minor version number of the linker.

SizeOfCode: Size of the code (text) section in the executable.
SizeOfInitializedData: Size of the initialized data section.
SizeOfUninitializedData: Size of the uninitialized data section.
AddressOfEntryPoint: Memory address where execution starts.
BaseOfCode: Starting address of the code section in memory.
BaseOfData: Starting address of the data section in memory.
ImageBase: Preferred address of the first byte of the image in memory.
SectionAlignment: Alignment of sections in the memory, in bytes.
FileAlignment: Alignment of sections in the file, in bytes.
MajorOperatingSystemVersion: Major version number of the required operating system.
MinorOperatingSystemVersion: Minor version number of the operating system.
MajorImageVersion: Major version number of the image.
MinorImageVersion: Minor version number of the image.
MajorSubsystemVersion: Major version number of the subsystem.
MinorSubsystemVersion: Minor version number of the subsystem.
SizeOfImage: Total size of the image in memory, including all headers.
SizeOfHeaders: Combined size of all headers.
CheckSum: Image file checksum.
Subsystem: Subsystem required to run this image.
DllCharacteristics: DLL characteristics flags.
SizeOfStackReserve: Size of stack to reserve.
SizeOfStackCommit: Size of stack to commit.
SizeOfHeapReserve: Size of heap to reserve.
:**SizeOfHeapCommit:** Size of heap to commit.
LoaderFlags: Reserved, must be zero.
NumberOfRvaAndSizes: Number of data-directory entries in the remainder of the optional header.
SectionsNb: Number of sections in the executable.
SectionsMeanEntropy: Average entropy of sections, indicating randomness.
SectionsMinEntropy: Minimum entropy found in a section.
SectionsMaxEntropy: Maximum entropy in a section.
SectionsMeanRawsize: Average size of sections in the file.
SectionsMinRawsize: Minimum size of a section in the file.
SectionMaxRawsize: Maximum size of a section in the file.
SectionsMeanVirtualsize: Average virtual size of sections.
SectionsMinVirtualsize: Minimum virtual size of a section.
SectionMaxVirtualsize: Maximum virtual size of a section.
ImportsNbDLL: Number of imported DLLs.
ImportsNb: Total number of imports from all DLLs.
ImportsNbOrdinal: Number of imports using ordinals.
ExportNb: Number of exported symbols.
ResourcesNb: Number of resources in the executable.
ResourcesMeanEntropy: Average entropy of resources.
ResourcesMinEntropy: Minimum entropy of resources.
ResourcesMaxEntropy: Maximum entropy of resources.

ResourcesMeanSize: Average size of resources.
ResourcesMinSize: Minimum size of a resource.
ResourcesMaxSize: Maximum size of a resource.
LoadConfigurationSize: Size of the load configuration structure.
VersionInformationSize: Size of the version information structure.

Strategy

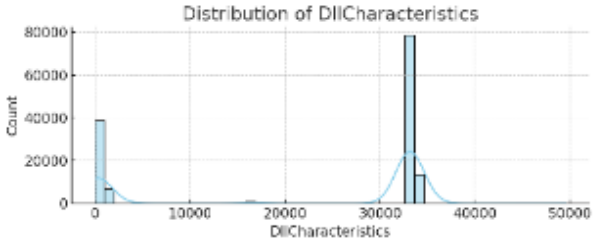
We perform an analysis of each column to determine the distribution of the data. The objective is to determine what normalization strategy might most accurately depict the information. We implement an algorithm that automatically factors in the general statistical tendencies identified in section Statistical Measures of Data of this document. Below is the result:

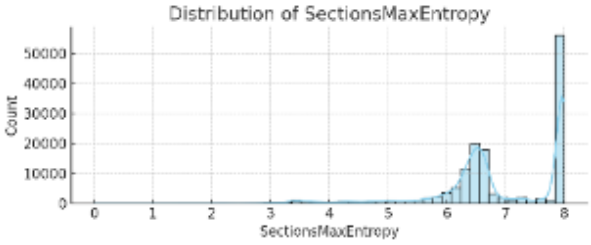
	Feature	Normalization Strategy
0	Machine	Robust Scaling
1	SizeOfOptionalHeader	Standard Scaling
2	Characteristics	Robust Scaling
3	MajorLinkerVersion	Standard Scaling
4	MinorLinkerVersion	Robust Scaling
5	SizeOfCode	Robust Scaling
6	SizeOfInitializedData	Robust Scaling
7	SizeOfUninitializedData	Robust Scaling
8	AddressOfEntryPoint	Robust Scaling
9	BaseOfCode	Robust Scaling
10	BaseOfData	Robust Scaling
11	ImageBase	Robust Scaling
12	SectionAlignment	Robust Scaling
13	FileAlignment	Robust Scaling
14	MajorOperatingSystemVersion	Robust Scaling
15	MinorOperatingSystemVersion	Robust Scaling
16	MajorImageVersion	Robust Scaling
17	MinorImageVersion	Robust Scaling
18	MajorSubsystemVersion	Standard Scaling
19	MinorSubsystemVersion	Robust Scaling
20	SizeOfImage	Robust Scaling
21	SizeOfHeaders	Robust Scaling
22	CheckSum	Robust Scaling
23	Subsystem	Standard Scaling
24	DllCharacteristics	Robust Scaling
25	SizeOfStackReserve	Standard Scaling
26	SizeOfStackCommit	Robust Scaling
27	SizeOfHeapReserve	Standard Scaling
28	SizeOfHeapCommit	Robust Scaling

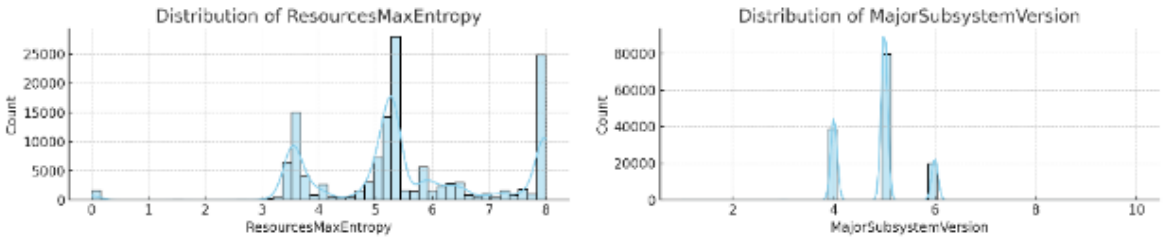
29	LoaderFlags	Robust Scaling
30	NumberOfRvaAndSizes	Robust Scaling
31	SectionsNb	Standard Scaling
32	SectionsMeanEntropy	Standard Scaling
33	SectionsMinEntropy	Standard Scaling
34	SectionsMaxEntropy	Standard Scaling
35	SectionsMeanRawsize	Robust Scaling
36	SectionsMinRawsize	Robust Scaling
37	SectionMaxRawsize	Robust Scaling
38	SectionsMeanVirtualsize	Robust Scaling
39	SectionsMinVirtualsize	Robust Scaling
40	SectionMaxVirtualsize	Robust Scaling
41	ImportsNbDLL	Standard Scaling
42	ImportsNb	Robust Scaling
43	ImportsNbOrdinal	Robust Scaling
44	ExportNb	Robust Scaling
45	ResourcesNb	Robust Scaling
46	ResourcesMeanEntropy	Standard Scaling
47	ResourcesMinEntropy	Standard Scaling
48	ResourcesMaxEntropy	Standard Scaling
49	ResourcesMeanSize	Robust Scaling
50	ResourcesMinSize	Robust Scaling
51	ResourcesMaxSize	Robust Scaling
52	LoadConfigurationSize	Robust Scaling
53	VersionInformationSize	Standard Scaling
54	legitimate	Robust Scaling

We observe most frequently a robust scaling strategy selection. This strategy is important when handling outliers, which is logical for this dataset where values are frequently outside the norm. The selection process was determined by an algorithm that determined rudimentarily whether to select either standard or robust scaling.

After having the algorithm evaluate the rudimentary tendency of the data, we consider from the column histograms which subsequent normalization strategy might be more useful. One common observation is a bimodal pattern across many columns, such as what can be observed below:







By human inspection of the data, we determine where to consider remapping bimodal distributions into simpler integer values such as binary or ternary.

We also consider eliminating metadata flags, such as machine, md5, etc., because they present a form of data leakage for ground truth. For instance, machine seems to indicate a strong correlation to the legitimate tag. This is not something that would be useable in practice. We believe that the reason for this correlation is because of the dependency on the dataset, where older samples of malware, with a specific machine value, are the primary composition of what we could find. On the other hand, legitimate programs that are more easily accessible don't encounter such a restriction, causing the models to simply perform a differentiation on the basis of the machine label unreliably.

We consider the employment of PCA or human inspection to determine which of the columns are least significant in the process of classification. In the case of human inspection, a suggestion is to reduce the identifying data down to 20 elements on the basis of the highest correlation.

In the case of PCA, we observe the following tendencies when reducing the dataset down to 20 dimensions:

```
        PC1       PC2       PC3       PC4       PC5       PC6       PC7  \
0  7.378535 -0.335818  0.572498 -1.472986  4.280163  0.131786  1.084550
1  0.387983 -0.062067  0.778378 -0.252464  0.186551 -0.031704 -0.255353
2 -0.595801 -0.041216  1.698929  0.509432 -1.260686 -0.052427 -0.521699
3 -0.980401 -0.039262  2.034930  0.723682 -1.865528 -0.074230 -0.666736
4 -0.522408 -0.046901  1.740208  0.028090 -0.348873 -0.054919 -0.474299

        PC8       PC9      PC10      PC11      PC12      PC13      PC14  \
0  1.955301  1.261469 -1.119162  2.055450  1.889068 -4.267056 -0.095221
1 -0.020428 -0.210345 -0.057190  0.301357  0.339883 -0.650556  0.081456
2 -0.365083 -0.414145 -0.126085  0.888783  0.170151 -0.824305 -0.046895
3 -0.598370 -0.584544 -0.064374  0.912201  0.285568 -0.836405 -0.002193
4 -0.139989 -0.258353 -0.150459  0.655235  0.075255 -0.887508  0.024859

       PC15      PC16      PC17      PC18      PC19      PC20
0  0.085302 -0.156855  4.289444 -4.842782  1.864886 -0.720424
1  0.084082 -0.444794  0.678285 -0.404935 -0.118682 -0.069978
2 -0.316627 -0.297171  0.651322 -0.100235 -0.144831 -0.071851
3 -0.271536 -0.330163  0.577299 -0.056644 -0.146924 -0.069758
```

```
4 -0.046808 -0.302674  0.546409 -0.286535 -0.170100 -0.088886
```

A correlation evaluation of those elements demonstrates the viability of PCA:

```
PC1: 0.359680
PC2: -0.344924
PC3: 0.230494
PC4: 0.503752
PC5: -0.018271
PC6: 0.001939
PC7: -0.000534
PC8: 0.405410
PC9: -0.039007
PC10: -0.014147
PC11: 0.616160
PC12: 0.133077
PC13: -0.087644
PC14: 0.005988
PC15: -0.039348
PC16: -0.020251
PC17: 0.072704
PC18: -0.027224
PC19: -0.014329
PC20: -0.008336
```

We observe the presence of elements with a high correlation in the PCA so, while useful, it is more difficult to interpret analytically. It's extremely likely that PCA in conjunction with our low-parameter models will not work as intended due to a generalization that is too broad for us to effectively form a conclusion. Further research is necessary for considering how PCA can be leveraged to form a clearer analysis, but for now it is discarded in favor of using the original metadata.

PCA is also particularly dangerous in its attempt to group the broad spectrum of malware into a singular cluster. It's important to consider implementing techniques of malware identification in future research, which is not effectively performed with a dataset consisting of reduced elements. Leaving the original metadata as the basis of training facilitates future endeavors in building more precise and specialized models for identifying particular forms of malware.